

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

# Кафедра системного програмування та спеціалізованих комп’ютерних систем

**Лабораторна робота №1**

з дисципліни

**«Бази даних і засоби управління»**

Виконала: студентка ІII курсу

ФПМ групи КВ-91

Мамай В.В.

Перевірив:

Петрашенко А. В.

Київ – 2021

**Лабораторна робота № 1.**

**Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL**

*Метою роботи* є здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL.

*Завдання* роботи полягає у наступному:

1. Розробити модель «сутність-зв’язок» предметної галузі, обраної студентом самостійно, відповідно до пункту «Вимоги до ER-моделі».
2. Перетворити розроблену модель у схему бази даних (таблиці) PostgreSQL.
3. Виконати нормалізацію схеми бази даних до третьої нормальної форми (3НФ).
4. Ознайомитись із інструментарієм PostgreSQL та pgAdmin 4 та внести декілька рядків даних у кожну з таблиць засобами pgAdmin 4.

Варіант (опис обраної предметної галузі):

Сервіс перегляду аніме серіалів.

Вимоги звітування щодо пунктів 1-4 завдання:

У звіті щодо пункту №1 завдання має бути:

* Перелік сутностей з описом їх призначення;
* Графічний файл розробленої моделі «сутність-зв’язок»;
* Назва нотації.

У звіті щодо пункту №2 завдання має бути:

* Опис процесу перетворення (наприклад, “сутність А було перетворено у таблицю А, а зв’язок R (M:N) зумовив появу додаткової таблиці R1 тощо);
* Схему бази даних у графічному вигляді з назвами таблиць (!) та зв’язками між ними.

У звіті щодо пункту №3 завдання має бути:

* Пояснення щодо відповідності схеми бази даних нормальним формам НФ1, НФ2 та НФ3. У випадку невідповідності надати опис необхідних змін у схемі;
* У випадку проведення змін у схемі бази даних надати оновлену версію схеми, інакше - не наводити схему.

У звіті щодо пункту №4 завдання має бути:

* Навести копії екрану з pgAdmin4, що відображають назви та типи стовпців (доступне у закладці “Columns” властивостей “Properties” таблиць дерева об’єктів у pgAdmin4);
* Навести копії екрану з pgAdmin4, що відображають вміст таблиць бази даних у PostgreSQL. Таблиці на зображенні обов'язково повинні мати назву!

**Звіт щодо пункту 1 завдання:**

Перелік сутностей з описом їх призначення:

Сутність Anime призначено для ідентифікації аніме серіалу, визначення кількості серій.

Сутність Genre призначено для визначення жанру аніме серіалів.

Сутність User призначено для ідентифікації користувача та надавання йому змоги дивитись та залишати відгуки до аніме.

Сутність Review призначено для залишання відгуків від користувачів до аніме серіалів.

Графічний файл розробленої моделі «сутність-зв‘язок»:

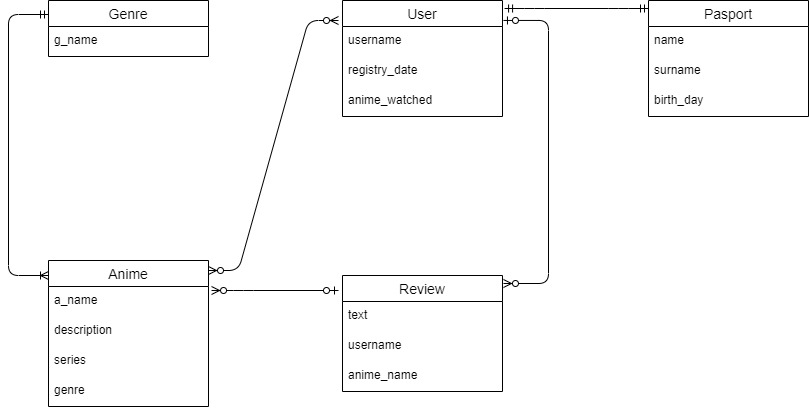
**

Рисунок 1 – Концептуальна модель

Назва нотації: нотація “Пташиної лапки (Crow’s foot)”

**Звіт щодо пункту 2 завдання:**

Опис процесу перетворення:

Зв’язок між сутностями Anime-User зумовив появу додаткової сутності watched.

Зв’язок між сутностями User-Pasport зумовив появу додаткової сутності user\_pasport.

Схема бази даних у графічному вигляді:

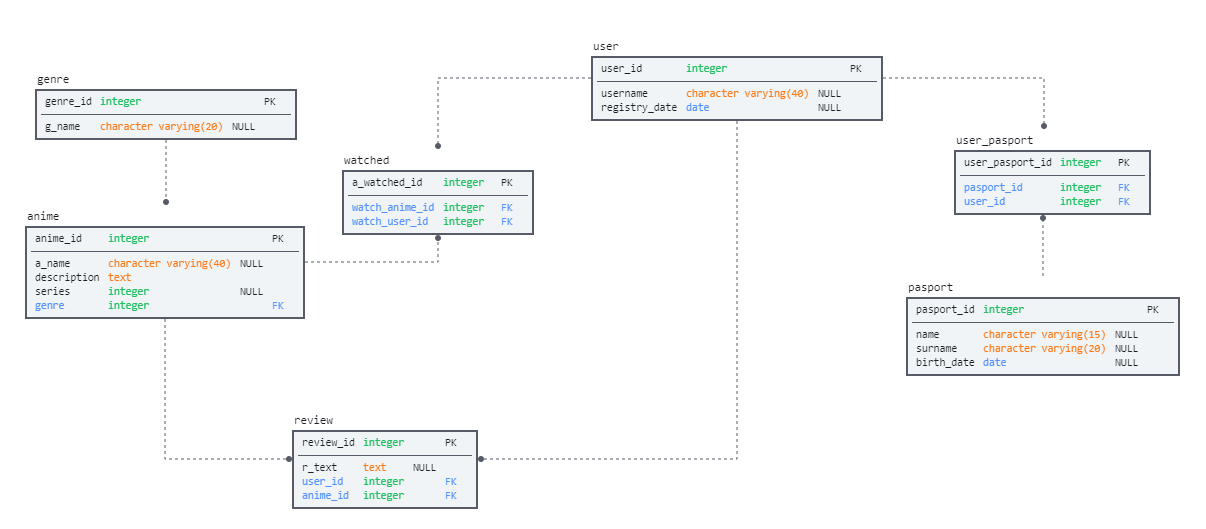


Рисунок 2 – Логічна модель

Структура БД «Сервіс перегляду аніме серіалів»

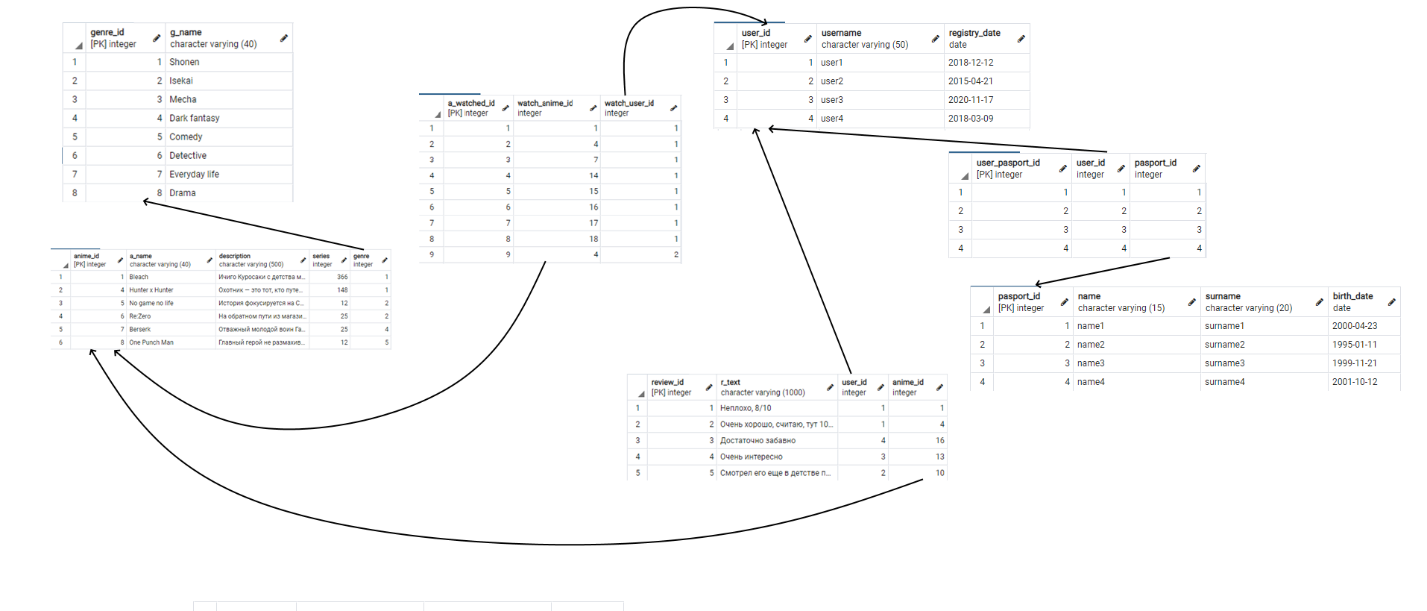


Рисунок 3 - структура БД «Сервіс перегляду аніме серіалів»

Опис структури БД «Сервіс перегляду аніме серіалів»

|  |  |  |
| --- | --- | --- |
| ВІДНОШЕННЯ | АТРИБУТ | ТИП |
| Genre  Вміщує інформацію про жанр | genre\_id(PK) – унікальний ID жанру в БД  g\_name – назва жанру | Числовий  Текст(20) |
| Anime  Вміщує інформацію про аніме | anime\_id(PK) – унікальний ID аніме в БД  a\_name – назва аніме  description – опис аніме  series – кількість серій аніме  genre(FK) – жанр даного аніме | Числовий  Текстовий(40)  Текстовий(1000)  Числовий  Числовий |
| User  Вміщує інформацію про користувача | user\_id(PK) – унікальний ID користувача в БД  username – нікнейм користувача  registry\_date - дата регістрації користувача  user\_pasport\_id(FK) – ID паспорта користувача | Числовий  Текстовий(40)  Дата  Числовий |
| Watched  Вміщує інформацію про переглянуті користувачами аніме | a\_watched\_id(PK) – унікальний ID перегляду  watch\_anime\_id(FK) – ID переглянутого аніме  watch\_user\_id(FK) – ID користувача, що переглянув аніме | Числовий  Числовий  Числовий |
| Review  Вміщує інформацію про відгуки до аніме | review\_id(PK) – унікальний ID відгуку  r\_text – текст відгуку  user\_id(FK) – ID користувача, що залишив відгук  anime\_id(FK) – ID аніме, до якого залишили відгук | Числовий  Текстовий(1000)  Числовий  Числовий |
| Pasport  Вміщує інформацію про паспорт користувача | pasport\_id(PK) – унікальній ID паспорта  name – ім’я користувача  surname – прізвище користувача  birth\_date – дата народження користувача | Числовий  Текстовий(15)  Текстовий(20)  Дата |
| User\_pasport  Зв‘язує таблиці User та Pasport | user\_pasport\_id(PK) – ідентифікатор запису  passport\_id(FK) – ідентифікатор паспорта  user\_id(FK) – ідентифікатор користувача | Числовий  Числовий  Числовий |

**Звіт щодо пункту 3 завдання:**

Пояснення щодо відповідності схеми бази даних нормальним формам:

Схеми бази даних відповідають вимогам 1НФ тому що дані в схемі атомарні тобто лише 1 елемент в кожній комірці.

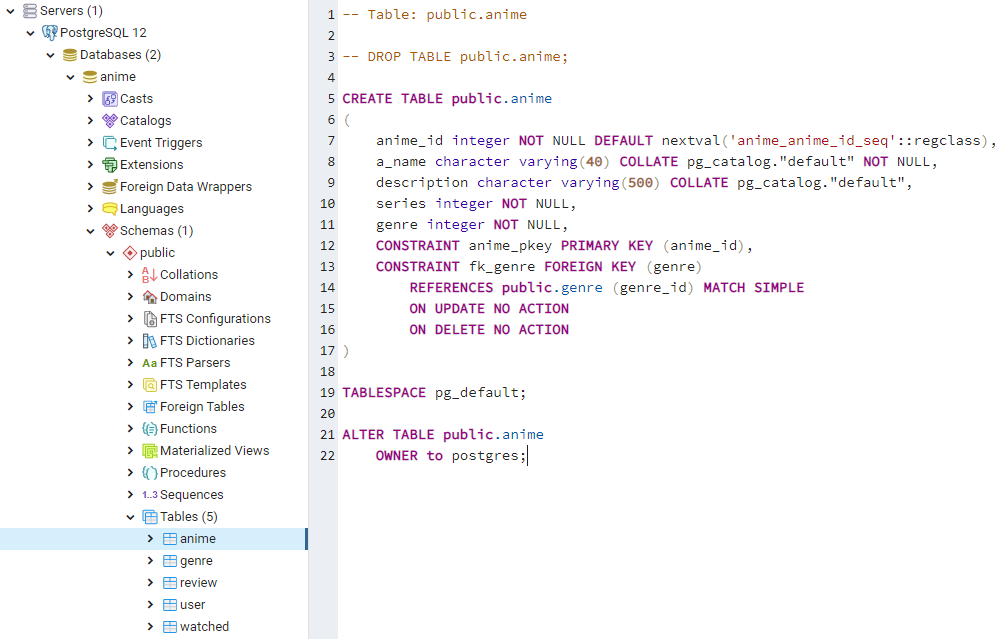
Схеми бази даних відповідають вимогам 2НФ тому що відповідають вимогам 1НФ та не мають потенціальних ключів, які складаються з декількох атрибутів.

Схеми бази даних відповідають вимогам 3НФ тому що відповідають 2НФ та відсутні транзитивні функціональні залежності неключових атрибутів від ключових.

**Звіт щодо пункту 4 завдання:**

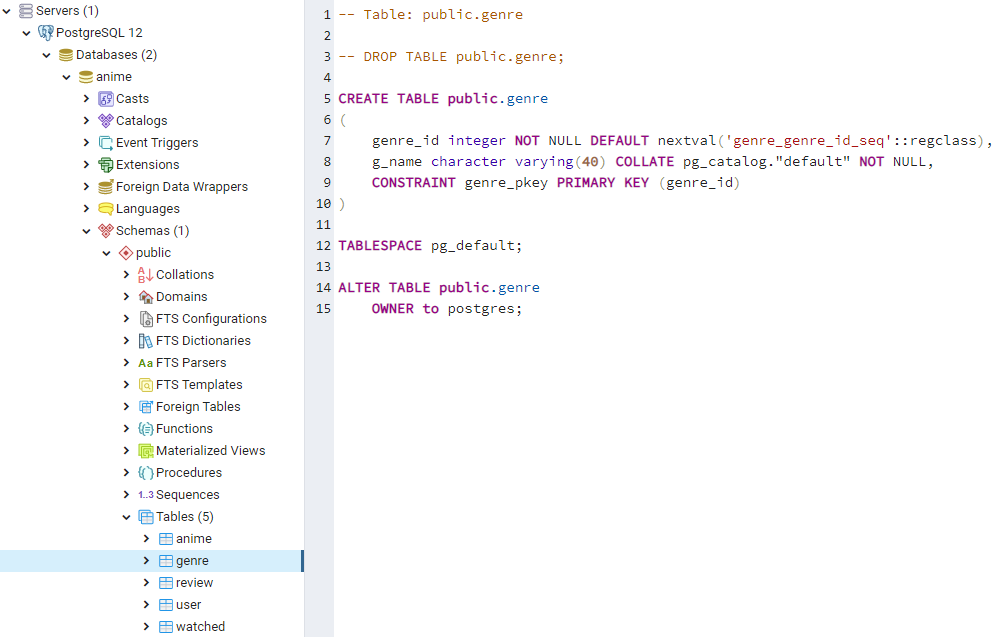
Копії екрану з pgAdmin4, що відображають назви та типи стовпців та їх вміст бази даних у PostgreSQL:

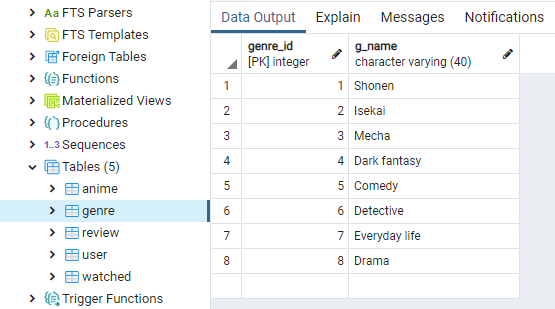
Таблиця anime:

******

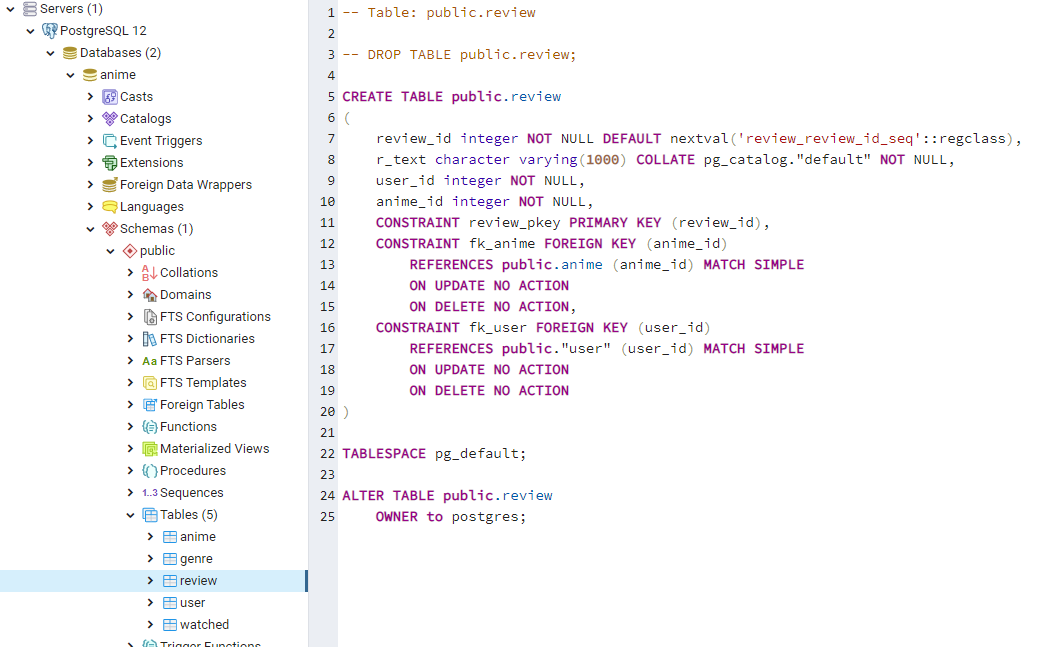
******

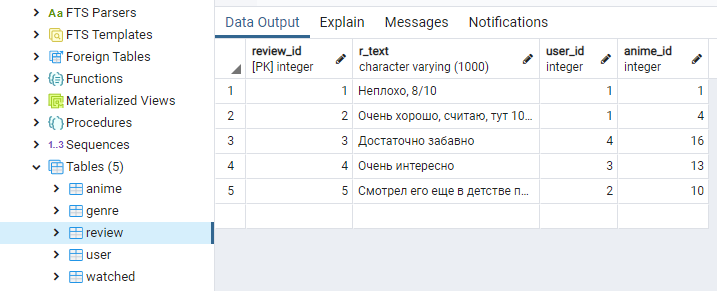
Таблиця genre:

****

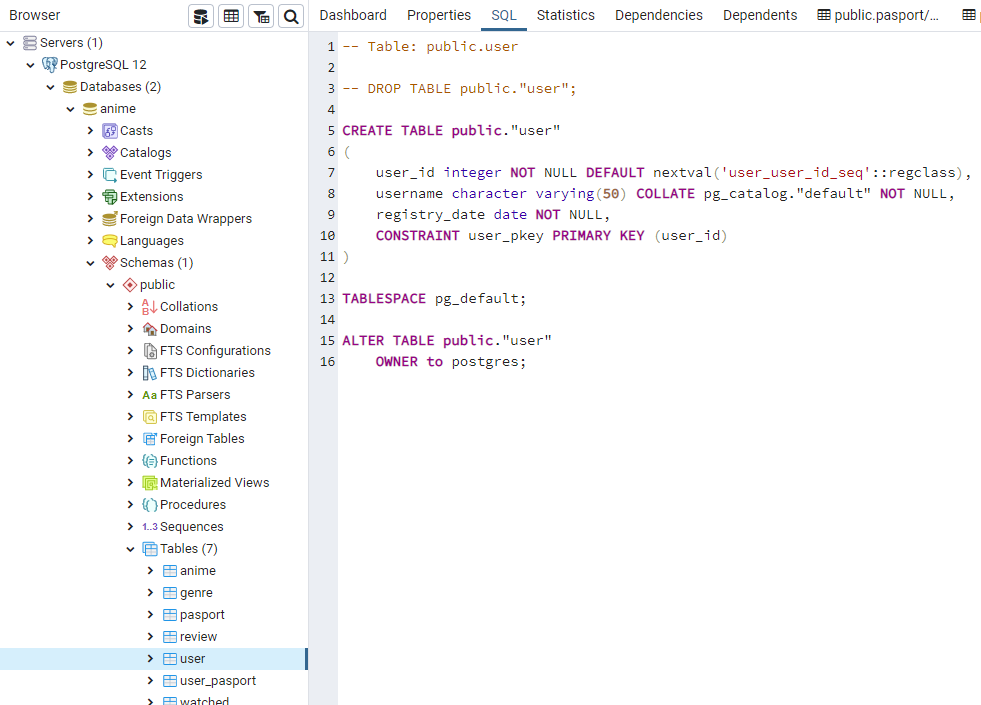
****

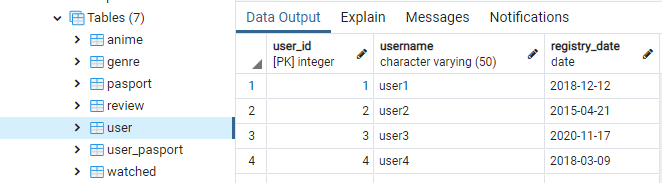
Таблиця review:

****

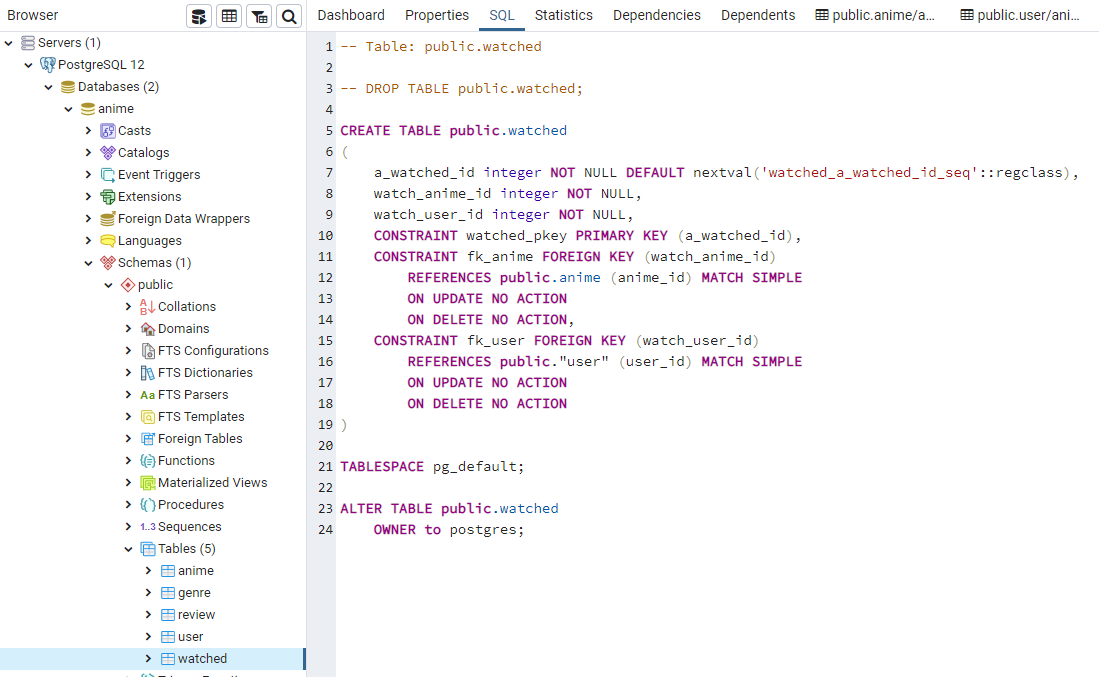
****

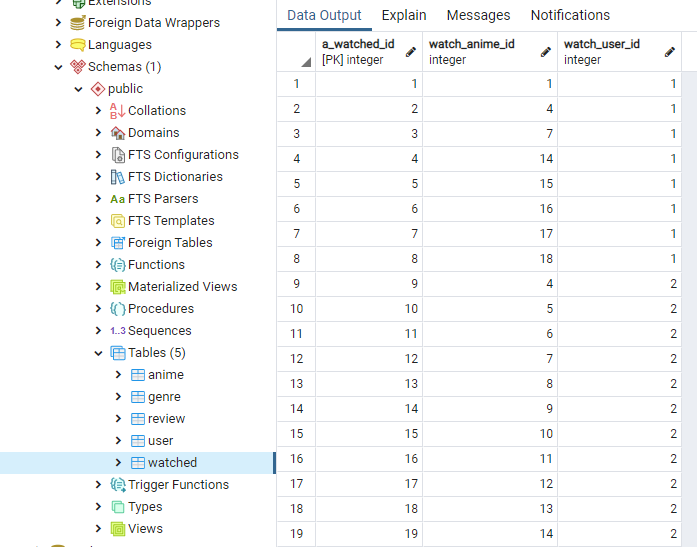
Таблиця user:

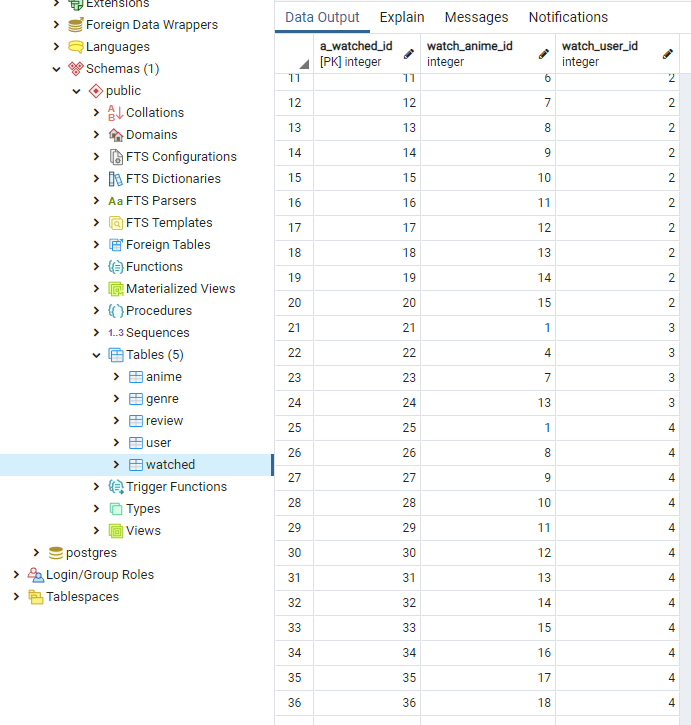
****

****

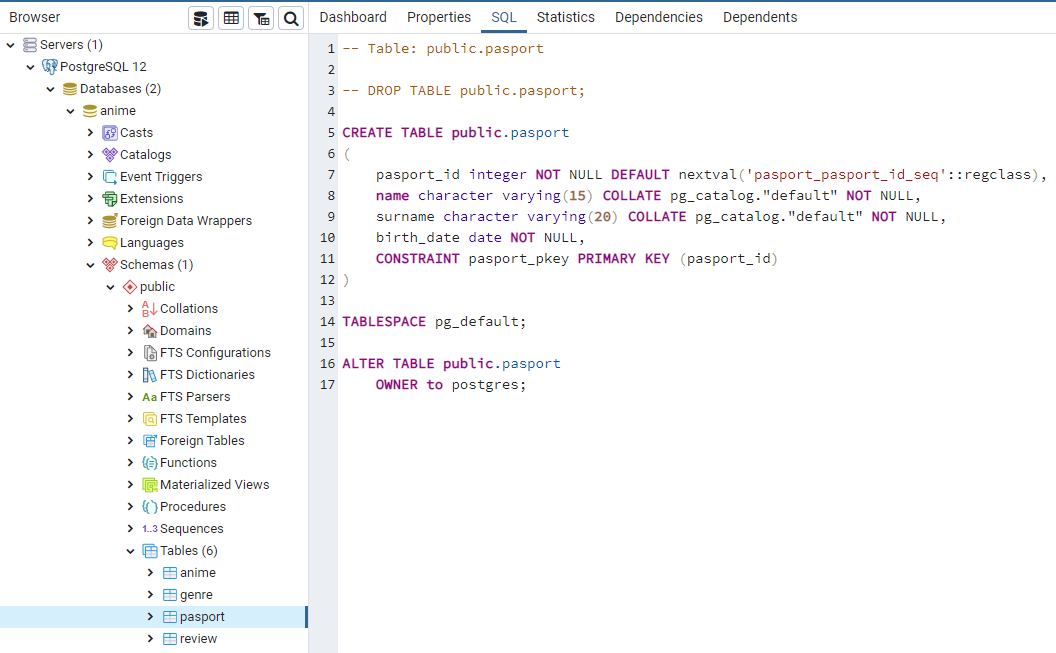
Таблиця watched:

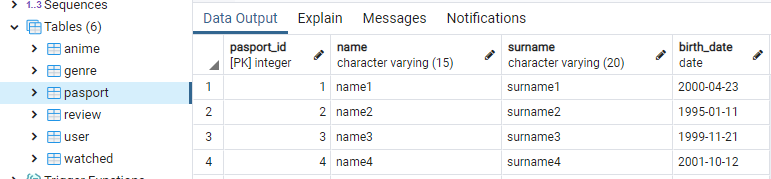
****

****

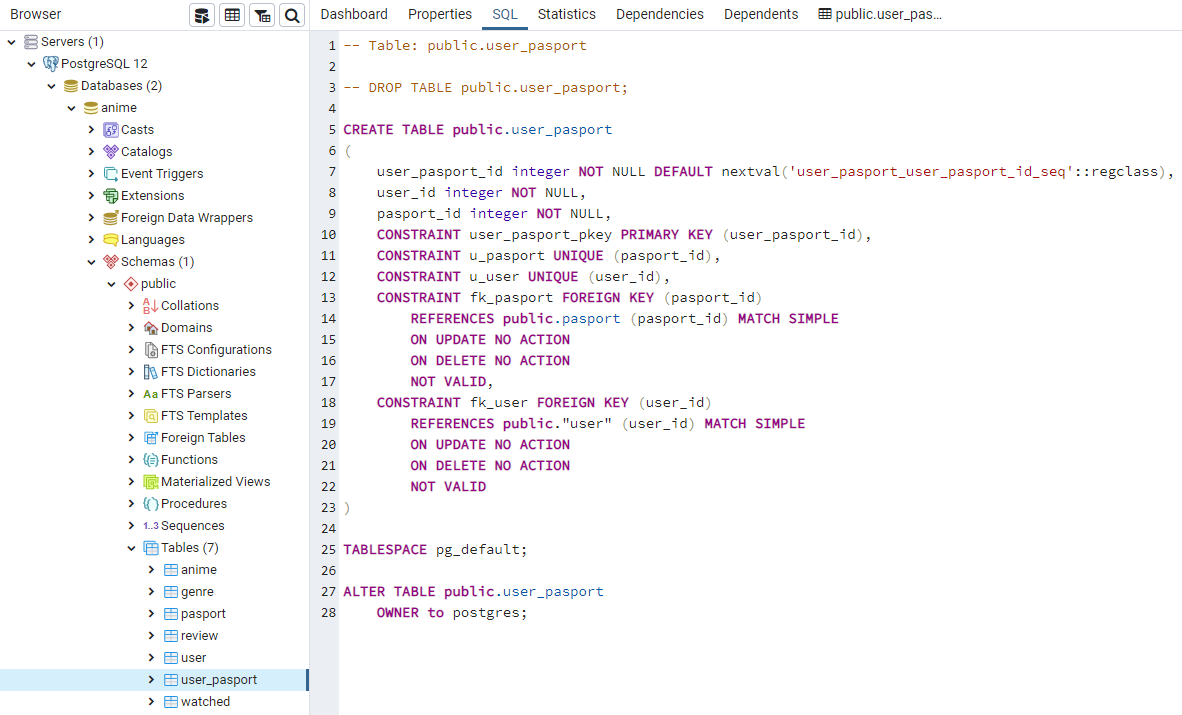
****

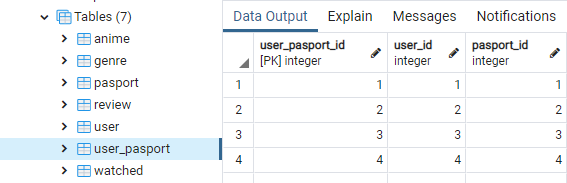
Таблиця pasport:

****

****

Таблиця user\_pasport:

****

****